

WHAT IS CLAIMED IS:

1 1. A method of bypassing an infrequent null pointer condition when compiling a
2 source program comprised of:
3 creating a fault to target translation table of the infrequent null pointer condition;
4 relating the infrequent null pointer condition to a procedural instruction in the fault to
5 target translation table; and
6 compiling the source program to an executable program.

1 2. The method of claim 1 further comprising:
2 gathering statistics as to the number of occurrences the infrequent null pointer
3 condition occurs;
4 determining an acceptable rate of occurrence; and
5 entering the infrequent condition into the fault to target translation table if the
6 infrequent null pointer condition does not exceed the acceptable rate of
7 occurrence.

1 3. The method of claim 1 further comprising:
2 passing fault to target translation data from the fault to target translation table to the
3 compiler using a handler program.

1 4. The method of claim 2 further comprising:
2 passing fault to target translation data from the fault to target translation table to the
3 compiler using a handler program.

1 5. The method of claim 1 further comprising:
2 accessing the fault to target translation table during compiling of the source program.

1 6. The method of claim 2 further comprising:
2 accessing the fault to target translation table during compiling of the source program.

1 7. The method of claim 3 further comprising:
2 accessing the fault to target translation table during compiling of the source program.

1 8. The method of claim 4 further comprising:
2 accessing the fault to target translation table during compiling of the source program.

1 9. A computing system capable of bypassing an infrequent null pointer condition
2 when compiling a source program comprising:

3 a processor;
4 a computer readable medium coupled to the processor; and
5 computer code, encoded in the computer readable medium, configured to cause the
6 processor to:
7 create a fault to target translation table of the infrequent null pointer condition;
8 relate the infrequent null pointer condition to a procedural instruction in the
9 fault to target translation table; and
10 compile the source program to an executable program.

1 10. The computing system of claim 9 wherein the processor is further configured
2 to:
3 gather statistics as to the number of occurrences the infrequent null pointer
4 condition occurs;
5 determine an acceptable rate of occurrence; and
6 enter the infrequent condition into the fault to target translation table if the
7 infrequent null pointer condition does not exceed the acceptable rate of
8 occurrence.

1 11. The computing system of claim 9 wherein the processor is further configured
2 to:
3 pass fault to target translation data from the fault to target translation table to
4 the compiler using a handler program.

1 12. The computing system of claim 10 wherein the processor is further configured
2 to:
3 pass fault to target translation data from the fault to target translation table to
4 the compiler using a handler program.

1 13. The computing system of claim 9 wherein the processor is further configured
2 to:
3 access the fault to target translation table during compiling of the source
4 program.

1 14. The computing system of claim 10 wherein the processor is further configured
2 to:
3 access the fault to target translation table during compiling of the source
4 program.

1 15. The computing system of claim 11 wherein the processor is further configured
2 to:
3 access the fault to target translation table during compiling of the source
4 program.

1 16. The computing system of claim 12 wherein the processor is further configured
2 to:
3 access the fault to target translation table during compiling of the source
4 program.

1 17. An apparatus to bypass an infrequent null pointer condition when compiling a
2 source program comprised of:
3 means for creating a fault to target translation table of the infrequent null pointer
4 condition;
5 means for relating the infrequent null pointer condition to a procedural instruction in
6 the fault to target translation table; and
7 means for compiling the source program to an executable program.

1 18. The apparatus of claim 17 further comprised of:
2 means for gathering statistics as to the number of occurrences the infrequent null
3 pointer condition occurs;
4 means for determining an acceptable rate of occurrence; and
5 means for entering the infrequent condition into the fault to target translation table if
6 the infrequent null pointer condition does not exceed the acceptable rate of
7 occurrence.

1 19. The apparatus of claim 17 further comprised of:
2 means for passing fault to target translation data from the fault to target translation
3 table to the compiler using a handler program.

1 20. The apparatus of claim 18 further comprised of:
2 means for passing fault to target translation data from the fault to target translation
3 table to the compiler using a handler program.

1 21. The apparatus of claim 17 further comprised of:
2 means for accessing the fault to target translation table during compiling of the source
3 program.

1 22. The apparatus of claim 18 further comprised of:
2 means for accessing the fault to target translation table during compiling of the source
3 program.

1 23. The apparatus of claim 19 further comprised of:
2 means for accessing the fault to target translation table during compiling of the source
3 program.

1 24. The apparatus of claim 20 further comprised of:
2 means for accessing the fault to target translation table during compiling of the source
3 program.

1 25. A computer program product that bypasses an infrequent null pointer
2 condition when compiling a source program comprising:
3 a first set of instructions, executable on a computer system, configured to gather
4 statistics as to the number of occurrences the infrequent null pointer condition
5 occurs;
6 a second set of instructions, executable on the computer system, configured to
7 determine an acceptable rate of occurrence; and
8 a third set of instruction, executable on the computer system, configured to enter the
9 infrequent condition into the fault to target translation table if the infrequent
10 null pointer condition does not exceed the acceptable rate of occurrence.

11 26. The computer program product of claim 25 further comprising:
12 a fourth set of instructions, executable on the computer system, configured to gather
13 statistics as to the number of occurrences the infrequent null pointer condition
14 occurs;
15 a fifth set of instructions, executable on the computer system, configured to determine
16 an acceptable rate of occurrence; and
17 a sixth set of instructions, executable on the computer system, configured to enter the
18 infrequent condition into the fault to target translation table if the infrequent
19 null pointer condition does not exceed the acceptable rate of occurrence.

1 27. The computer program product of claim 25 further comprising:
2 a seventh set of instructions, executable on the computer system, configured to pass
3 fault to target translation data from the fault to target translation table to the
4 compiler using a handler program.

1 28. The computer program product of claim 26 further comprising:
2 a seventh set of instructions, executable on the computer system, configured to pass
3 fault to target translation data from the fault to target translation table to the
4 compiler using a handler program.

1 29. The computer program product of claim 25 further comprising:
2 an eighth set of instructions, executable on the computer system, configured to access
3 the fault to target translation table during compiling of the source program.

1 30. The computer program product of claim 26 further comprising:
2 an eighth set of instructions, executable on the computer system, configured to access
3 the fault to target translation table during compiling of the source program.

1 31. The computer program product of claim 27 further comprising:
2 an eighth set of instructions, executable on the computer system, configured to access
3 the fault to target translation table during compiling of the source program.

1 32. The computer program product of claim 28 further comprising:
2 an eighth set of instructions, executable on the computer system, configured to access
3 the fault to target translation table during compiling of the source program.